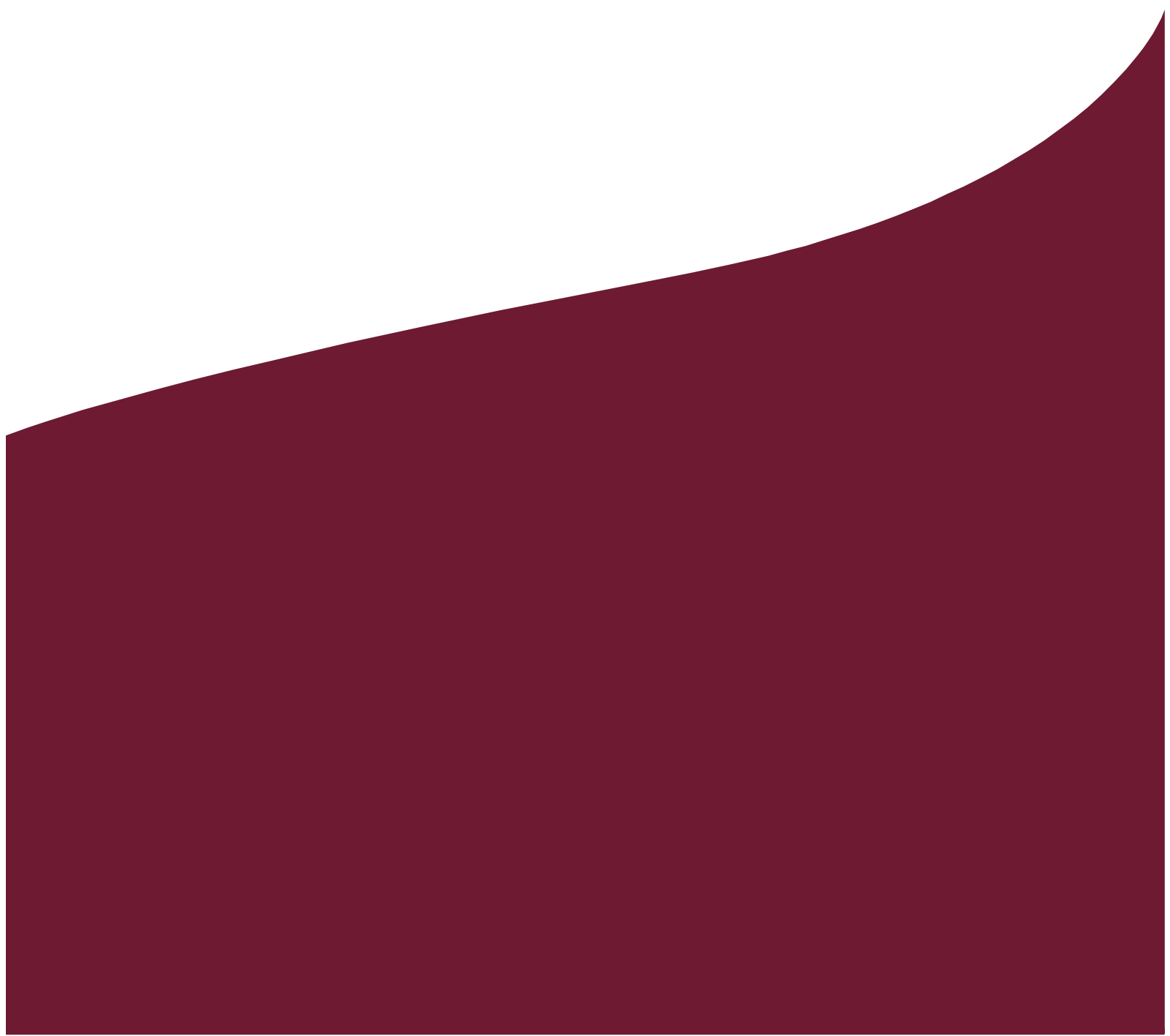


# Graphical UI Testing Checklist



# Table of Contents

---

**Prepared by:**

Atul Waghmare

---

**Submitted for:**

<<<text>>>

---

**Date:**

March 2009

|    |                                    |    |
|----|------------------------------------|----|
| 1  | What is GUI .....                  | 3  |
| 2  | Brief History of GUI.....          | 4  |
| 3  | GUI Validation Testing.....        | 5  |
| 4  | TextBox .....                      | 6  |
| 5  | Dropdown box/Combo Box .....       | 8  |
| 6  | Button .....                       | 10 |
| 7  | CheckBox .....                     | 11 |
| 8  | Radio Button/Options .....         | 12 |
| 9  | Browser Button .....               | 13 |
| 10 | Grid Control .....                 | 14 |
| 11 | Hyperlink.....                     | 15 |
| 12 | Image.....                         | 16 |
| 13 | Input Field Validations .....      | 17 |
| 14 | General Conditions to verify ..... | 18 |

## 1 What is GUI:

- A GUI as we now know is a computer-interface that uses images as well as typed text, with icons on the screen replacing many of the functions of the keyboard.
- Many sighted people find GUI easier to use, because they don't have to remember or to look up special commands for each program function.
- Less time is spent figuring out how to get the computer to do what you want it to do.
- People who are blind and visually impaired can use GUI, provided they have a reliable screen reader to translate what's on the screen into Braille or synthesized speech.

## 2 Brief History of GUI:

The history of GUI dates back to 1970's when the Project Smalltalk was established at Xerox Palo Alto Research Centre (Parc), which attempted to look into the future. The idea was to assume that in the future, computing power would be abundant and inexpensive and the best use of this available power will be made. Two influential developments resulted namely, Object-oriented Programming & the Graphical User Interface. Soon Apple took the track of Xerox. Apple came up with Lisa, the then, very powerful microcomputer that had a GUI. Macintosh was released in 1984, which consisted of a keyboard and mouse attached to a box containing a 9-inch monochrome screen & a single floppy disc drive. Then came the "Big" Mac with 512 kB. At that time, the style of interface was usually referred to as "Wimp", derived from the components "window, icon, menus, and pointer". GUI – or even "gooie" very soon, replaced this name.

### 3 GUI Validation Testing:

GUI Validation Testing is basically required to ensure that the standards to be implemented during the development of each product are strictly taken care of while coding. It is mainly to verify the compliance of a company's own GUI Standards seeded in its products.

## 4 TextBox:

1. Check for enable and disable property as per the requirement.
2. **Disabled TextBox:**
  1. Check if any pre-populated value should be displayed as per requirement.
  2. Check if you cannot edit disabled TextBox.
  3. Check for conditions when that TextBox should be enabled.
  4. Disabled textbox should not accept any value.
  5. Check for cursor should not be displayed in disabled TextBox.
  6. Disabled TextBox should be grayed out.
3. **Enabled TextBox:**
  1. Check for pre-population of any value or BLANK value as per requirement.
  2. Check for blinking cursor when get focused.
  3. Check for RED asterisk for mandatory fields. (as per requirements)
  4. Enter continuous characters still it stops to enter any character and measure width of the TextBox. It should stop you and width should be as per requirement.
  5. Check for spaces and its combination with any characters. It should not allow only spaces.
  6. Check for numbers and special characters like 1, 2, \$, #, +, - etc.
  7. Check for combination of alphabets, numeric, signs and special characters.
  8. Enter valid value in mandatory TextBox and save the window. Again come to same window and delete that value and test if it asks for value as it is mandatory.
  9. If value is coming from DB then check if you can edit same.
  10. If value is coming from DB then check if the value is correct and populating correctly.
  11. Check if DB accepting "Redundant" values.
  12. Check if correct "Text" associated with TextBox.
  13. Check for alignment of TextBox and Text associated with TextBox are proper as per other controls on window.

14. Check for “Formatted TextBox” which should have proper format pre-defined and accepting the same number characters in the same format for ex: SSN number should accept XXX-XXX-XXX in this format and - - characters should be displayed in TextBox before entering anything.
15. Check if “formatted” TextBox are not accepting less than the allocated characters.
16. Check if “Browser” button is associated with TextBox then same values which selected in “Browser” should be displayed and in its proper format. Also check for edit function for same as per business functionality.
17. Check for “Scrolling” option available if it accepting characters out of boundary. Also check for each Up and Down scroll is working properly.
18. If it has an option like after clicking on checkbox the TextBox should be disabled then check if already entered characters are get wiped off and TextBox should be disabled if clicked on such checkbox.
19. SHIFT and Arrow should Select Characters. Selection should also be possible with mouse. Double Click should select all text in box.
20. Move the Mouse Cursor over all Enterable Text Boxes. Cursor should change from arrow to Insert Bar.

## 5 Dropdown box/Combo Box:

### 1. Only select list control:

1. Check for appropriate sign displayed (as arrow) over control.
2. Check for default value that should be displayed or it should be BLANK.
3. Check if mandatory field can accept BLANK value.
4. Check if it should accept numbers or special characters apart from list.
5. Click on the arrow of the control and check if list should be displayed.
6. Check if list is not empty.
7. Check if list contains number of values then “Scroll” option should display and it should be working.
8. Check entering first letter of option in control and value should get selected.
9. Try if we cannot delete selected option in control. We should not be able to delete.
10. Check for alignment of list is associated with control.
11. Check for “TEXT” identifier displayed with the control.
12. Check after clicking on arrow image should display as clicked.
13. Are all the dropdowns on this screen sorted correctly? Alphabetic sorting is the default unless otherwise specified.
14. In drop down list boxes, ensure that the names are not abbreviations / cut short

### 2. Select list and User entry control:

1. Check for appropriate sign displayed (as arrow and bubble) over control.
2. Check for default value that should be displayed or it should be BLANK.
3. Check if mandatory field can accept BLANK value.
4. Check if it should accept numbers or special characters apart from list.
5. Click on the arrow of the control and check if list should be displayed.
6. Check if list is not empty.
7. Check if list contains number of values then “Scroll” option should display and it should be working.
8. Check entering first letter of option in control and value should get selected.
9. Try if we cannot delete selected option in control. We should be able to delete.



10. Check if we can ENTER the text in this control. Check for alpha numeric and special characters like @\$ etc.
11. Check if we can select value from list when previously entered text is in the control.
12. Check if we can enter text if previously selected value is available in control.
13. Check for alignment of list is associated with control.
14. Check for “TEXT” identifier displayed with the control.
15. Check after clicking on arrow image should display as clicked.
16. Check entering same value as in list in the control then it should not handle it as a different value.

## 6 Button:

1. Check Enable or disable property of the control.
2. Verify Tab sequence can select control in respective manner.
3. Check change image after clicking event encounter.
4. Check alignment with other controls on the window.
5. Check appropriate TEXT associated with the control.
6. Check event occur after click of the control.
7. Check appropriate naming convention used for label of the control.
8. Check label of the control should not be editable.
9. Check the events with single click, Double click and more than double click of the control.
10. Check control should be displayed as “Highlighted” with hovering of mouse cursor on the control.
11. Check if appropriate information should display as per requirement for hovering of mouse cursor.
12. “Highlighted” button control should accept click event with keyboard “Enter” key.
13. Button should be activated with ALT+ character underlined in label of that button.
14. SPACE key from keyboard should activate button which is highlighted.
15. One button on screen should be by default with thick border and should activate when enter key pressed from keyboard at any time.
16. “Cancel” button should be activated when Esc key pressed from keyboard.
17. Make sure there isn’t duplication of underlined character of different buttons.
18. If pressing the Command button results in uncorrectable data e.g. closing an action step, there should be a message phrased positively with Yes/No answers where yes results in the completion of the action.
19. If Command Button leads to another Screen, and if the user can enter or change details on the other screen then the Text on the button should be followed by three dots. (Click...)
20. There should be a default button specified on the screen.
21. Default button should work correctly.
22. Verify that command buttons are all of similar size and shape, and same font & font

## 7 CheckBox:

1. Check enable and disabled property of the control
2. Check default value either checked or unchecked.
3. Verify appropriate label for each checkbox.
4. Check Tab sequence should select checkbox control in series.
5. Check mouse click should change property of checkbox from checked to unchecked or vice versa.
6. Check space bar key press from keyboard should change property of checkbox from checked to unchecked or vice versa.
7. Verify appropriate events associated with checkbox property change.
8. Check disabled checkbox should not be editable.
9. Check Enter key press from keyboard should change property of checkbox from checked to unchecked or vice versa.
10. Check alignment should be proper with other controls in window.
11. Control should be selected and checked or unchecked event should takes place when ALT+underlined character pressed from keyboard of the appropriate checkbox.

## 8 Radio Button/Options:

1. Check for default radio button selection.
  2. Check if we cannot select more than one radio buttons.
  3. Verify appropriate label for each radio button.
  4. Check Tab sequence should select radio button control in series.
  5. Check mouse click should change property of radio button from checked to unchecked or vice versa.
  6. Check space bar key press from keyboard should change property of radio button from checked to unchecked or vice versa.
  7. Verify appropriate events associated with radio button property change.
  8. Check disabled radio button should not be editable.
  9. Check Enter key press from keyboard should change property of radio button from checked to unchecked or vice versa.
  10. Check alignment should be proper with other controls in window.
  11. Control should be selected and checked or unchecked event should takes place when ALT + underlined character pressed from keyboard of the appropriate radio button.
  12. If a set of radio buttons represent a fixed set of values such as A, B and C then what happens if a blank value is retrieved from the database? (In some situations rows can be created on the database by other functions which are not screen based and thus the required initial values can be incorrect.)
- Limitations of Sanity Testing

## 9 Browser Button:

1. Check for default value in browser textbox.
2. Check if we can edit the default value.
3. Verify appropriate label for browser control.
4. Check Tab sequence should select browser button control in series.
5. Check mouse hovering should change color of browser button.
6. Check space bar key press from keyboard should change select browser button and should click event occur.
7. Verify appropriate events associated with browser button click.
8. Check disabled browser button should not be clickable.
9. Check Enter key press from keyboard should click browser button and event should occur.
10. Check alignment should be proper with other controls in window.
11. Control should be selected and event should takes place when ALT + underlined character pressed from keyboard of the appropriate browser button.

## 10 Grid Control:

1. Check for default value in grid control.
2. Check if we can edit the default value.
3. Verify appropriate label for each column in grid control and a heading to entire grid control.
4. Check Tab sequence should select each row in grid control in series.
5. Check space bar key press from keyboard should change only color of row in grid control and should not do click event.
6. Verify appropriate events associated with grid control row double click.
7. Check disabled grid control should not be clickable.
8. Check Enter key press from keyboard should not click grid control and event should not occur.
9. Check alignment should be proper with other controls in window.
10. Sorting in ascending and descending with respect to each column should occur on click of the column name with appropriate triangle should be seen.
11. Check if scroll is performing well with arrow key from keyboard, wheel of mouse and click event of scroll buttons on control.

## 11 Hyperlink:

1. Hyperlink should have standard color to identify.
2. Hyperlink should be underlined to each character.
3. Try to open hyperlink with single and double click of mouse.
4. Try to open hyperlink with “Enter” key of keyboard.
5. Try to open hyperlink with spacebar button of keyboard.
6. Ensure Tab sequence counts particular hyperlink.
7. Try to open hyperlink with CTRL + Enter key and CTRL + mouse click.
8. Verify correct webpage opens with given hyperlink.

## 12 Image:

1. Check image quality should not be blurring.
2. Verify if zoom in and zoom out option available.
3. Verify same image should get displayed either by scan or by import.
4. Verify rotation of the image.



## 13 Input Field Validations:

### 1. Input Number:

1. Verify that lowest and highest values are handled correctly
2. Verify that invalid values are logged and reported
3. Verify that valid values are handles by the correct procedure
4. Verify that numeric fields with a blank in position 1 are processed or reported as an error
5. Verify that fields with a blank in the last position are processed or reported as an error an error
6. Verify that both + and - values are correctly processed
7. Verify that division by zero does not occur
8. Include value zero in all calculations
9. Include at least one in-range value
10. Include maximum and minimum range values
11. Include out of range values above the maximum and below the minimum
12. Verify that upper and lower values in ranges are handled correctly

### 2. Input Date:

1. Verify that leap years are validated correctly & do not cause errors/miscalculations
2. Verify that month code 00 and 13 are validated correctly & do not cause errors/miscalculations
3. Verify that month values 00 and 13 are reported as errors
4. Verify that day values 00 and 32 are validated correctly & do not cause errors/miscalculations
5. Verify that Feb. 28, 29, 30 are validated correctly & do not cause errors/miscalculations
6. Verify that Feb. 30 is reported as an error
7. Verify that century change is validated correctly & does not cause errors/miscalculations
8. Verify that out of cycle dates are validated correctly & do not cause errors/miscalculations

### 3. Input Alpha fields:

1. Verify for blank and non-blank data
2. Verify Including lowest and highest values
3. Verify Including invalid characters & symbols
4. Verify Including valid characters
5. Verify Including data items with first position blank

## 14 General Conditions to verify:

1. Failure of validation on every field should cause a sensible user error message.
2. If the user enters an invalid value and clicks on the OK button (i.e. does not TAB off the field) the invalid entry should be identified and highlighted correctly with an error message.
3. For all numeric fields check whether negative numbers can and should be able to be entered.
4. If any of the database columns don't allow null values then the corresponding screen fields must be mandatory. (If any field which initially was mandatory has become optional then check whether null values are allowed in this field.)
5. Have the menu options which apply to your screen got fast keys associated and should they have?
6. Does the Tab Order specified on the screen go in sequence from Top Left to bottom right? This is the default unless otherwise specified.
7. Are all read-only fields avoided in the TAB sequence?
8. Are all disabled fields avoided in the TAB sequence?
9. Is the cursor positioned in the first input field or control when the screen is opened?
10. When an error message occurs does the focus return to the field in error when the user cancels it?
11. When the user Alt+Tab's to another application does this have any impact on the screen upon return to the application?
12. Is the data saved when the window is closed by double clicking on the close box?
13. Check the maximum field lengths to ensure that there are no truncated characters?
14. If a particular set of data is saved to the database check that each value gets saved fully to the database. i.e. beware of truncation (of strings) and rounding of numeric values.
15. Verify the existence of the "Help" menu.
16. Verify that all buttons on all tool bars have corresponding key commands.
17. Verify that each menu command has an alternative (hot-key) key sequence which will invoke it where appropriate.
18. Ensure that duplicate hot keys do not exist on each screen
19. Verify consistency of mouse actions across windows.
20. Verify that the screen/window does not have a cluttered appearance
21. Ctrl + F6 opens next tab within tabbed window
22. Shift + Ctrl + F6 opens previous tab within tabbed window

- 23. Alt+F4 will close the tabbed window and return you to main screen or previous screen (as appropriate), generating "changes will be lost" message if necessary.
- 24. Ensure all fields are disabled in read-only mode.
- 25. All text should be left-justified, followed by a colon tight to it.